

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2001年 3月21日

出 願 番 号

Application Number:

特願2001-080205

出 願 人

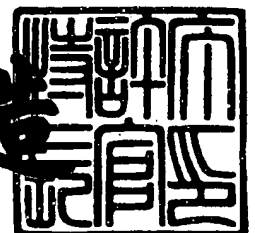
Applicant(s):

日本ビクター株式会社

2001年11月16日

特 許 庁 長 官  
Commissioner,  
Japan Patent Office

及 川 耕 造



【書類名】 特許願

【整理番号】 413000152

【提出日】 平成13年 3月21日

【あて先】 特許庁長官殿

【国際特許分類】 H03M 7/14  
G11B 20/14 341

【発明者】

【住所又は居所】 神奈川県横浜市神奈川区守屋町3丁目12番地 日本ビクター株式会社内

【氏名】 速水 淳

【特許出願人】

【識別番号】 000004329

【氏名又は名称】 日本ビクター株式会社

【代表者】 守随 武雄

【電話番号】 045-450-2423

【先の出願に基づく優先権主張】

【出願番号】 特願2001- 4628

【出願日】 平成13年 1月12日

【手数料の表示】

【予納台帳番号】 003654

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 変調方法、変調装置、復調方法、復調装置、情報記録媒体、情報伝送方法および情報伝送装置

【特許請求の範囲】

【請求項 1】

4 ビット単位の入力データ語を 6 ビット単位 of 出力符号語に符号化するための変換をする際に、前記各入力データ語に対応する前記各出力符号語と、次の前記入力データ語を符号化するために使用される符号化テーブルを指定する符号化テーブル指定情報とを含む複数の符号化テーブルを参照し、前記各出力符号語は 2 進数の出力符号語列として順次直接結合して (1, k) RLL (ラン・レンジス・リミテッド) 規則で k は 7 以上で 12 以下の何れかを満足する出力符号語として出力することを特徴とする変調方法。

【請求項 2】

前記複数の符号化テーブルは、少なくとも第 1, 第 2 符号化テーブルを有しており、所定の入力データ語に対応する前記第 1 符号化テーブル上の第 1 出力符号語と、前記所定の入力データ語と同一の入力データ語に対応する前記第 2 符号化テーブル上の第 2 出力符号語とをそれぞれ NRZI 変調した信号が逆極性であり、かつ、ある特定の出力符号語を出力した後に、前記第 1, 第 2 出力符号語のいずれを選択しても、選択された出力符号語は (1, k) RLL 規則で k は 7 以上で 12 以下の何れかを満足する出力符号語であることを特徴とする請求項 1 記載の変調方法。

【請求項 3】

前記第 1, 第 2 出力符号語のいずれかを選択する選択手段を備えたことを特徴とする請求項 2 記載の変調方法。

【請求項 4】

符号化テーブル指定情報で指定された入力データ語に対応する出力符号語が前記第 1, 第 2 出力符号語のいずれかである事を検出し、この検出結果に基づいて前記第 1, 第 2 符号化テーブルのいずれかを指定する符号化テーブル指定情報を前記複数の符号化テーブルに出力し、前記複数の符号化テーブルの中から指定

された符号化テーブルを用いて入力データ語に対応して順次出力される出力符号語を、出力符号語の極性毎に分別して蓄積し、指定された符号化テーブルから順次出力される出力符号語毎に、前記出力符号語メモリ手段にメモリされている出力符号語に対応するCDS（コードワード・デジタル・サム）を順次加算したDSV（デジタル・サム・バリエーション）を蓄積し、前記蓄積したDSVの絶対値の大きさを基に、出力符号語系列を選択して順次出力することを特徴とする請求項3に記載の変調方法。

#### 【請求項5】

特定の入力データ語に対して符号語ビットの最短反転が連続することを検出し、前記最小反転の所定数の連続を監視し、前記最小反転が所定の数だけ続いた場合、符号語ビットの最短反転所定内の連続数に収めることを特徴とする請求項1乃至請求項4 何れか1項に記載のkは8以上12以下の何れかを満足する変調方法。

#### 【請求項6】

4ビット単位の入力データ語を6ビット単位の出力符号語に符号化する変換手段を有し、前記変換手段は、前記入力データ語を前記出力符号語にそれぞれ符号化するための符号化テーブルを複数備えており、前記各符号化テーブルのそれぞれには前記各入力データ語に対応する前記各出力符号語と、次の前記入力データ語を符号化するために使用される符号化テーブルを指定する符号化テーブル指定情報とを含み、前記各出力符号語は2進数の出力符号語列として順次直接結合して(1, k) RLL（ラン・レンジス・リミテッド）規則でkは7以上で12以下の何れかを満足する出力符号語として出力することを特徴とする変調装置。

#### 【請求項7】

前記複数の符号化テーブルは、少なくとも第1, 第2符号化テーブルを備えており、所定の入力データ語に対応する前記第1符号化テーブル上の第1出力符号語と、前記所定の入力データ語と同一の入力データ語に対応する前記第2符号化テーブル上の第2出力符号語とをそれぞれNRZI変調した信号が逆極性であり、かつ、ある特定の出力符号語を出力した後に、前記第1, 第2出力符号語のいずれを選択しても、選択された出力符号語は(1, k) RLL規則でkは7以上

で 1 2 以下の何れかを満足する出力符号語であることを特徴とする請求項 6 記載の変調装置。

【請求項 8】

前記第 1, 第 2 出力符号語のいずれかを選択する選択手段を備えたことを特徴とする請求項 7 記載の変調装置。

【請求項 9】

符号化テーブル指定情報で指定された入力データ語に対応する出力符号語が前記第 1, 第 2 出力符号語のいずれかである事を検出し、この検出結果に基づいて前記第 1, 第 2 符号化テーブルのいずれかを指定する符号化テーブル指定情報を前記複数の符号化テーブルに出力する符号化テーブル指定手段と、前記複数の符号化テーブルの中から指定された符号化テーブルを用いて入力データ語に対応して順次出力される出力符号語を、出力符号語の極性毎に分別してメモリする出力符号語メモリ手段と、指定された符号化テーブルから順次出力される出力符号語毎に、前記出力符号語メモリ手段にメモリされている出力符号語に対応する C D S (コードワード・ディジタル・サム) を順次加算した D S V (ディジタル・サム・バリエーション) をメモリする D S V メモリ手段と、前記 D S V メモリ手段から出力される D S V の絶対値の大きさを基に、前記出力符号語メモリ手段から順次出力する出力符号語系列を選択する選択手段とを具備したことを特徴とする請求項 9 に記載の変調装置。

【請求項 1 0】

特定の入力データ語に対して符号語ビットの最短反転が連続することを検出する最小反転符号語検出手段と、前記最小反転の所定数の連続を監視する監視手段と、前記最小反転が所定の数だけ続いた場合、符号語ビットの最短反転所定内の連続数に収める符号語制御手段とを具備することを特徴とする請求項 6 乃至請求項 9 何れか 1 項に記載の k は 8 以上 1 2 以下の何れかを満足する変調装置。

【請求項 1 1】

請求項 1 乃至請求項 5 の何れか 1 項に記載の変調方法を用いて符号化された 6 ビット単位の符号語を連続化した符号語列を、再生データ列に復調する復調方法であって、前記符号語列を 6 ビット毎の符号語に再構成し、後続の符号語が前記

複数の符号化テーブルのうち、どの符号化テーブルで符号化がなされるかを示す判定情報と、後続の符号語とを基にして、前記符号語列を再生データ列に復調することを特徴とする復調方法。

【請求項 12】

請求項 6 乃至請求項 10 の何れか 1 項に記載の変調装置を用いて符号化された 6 ビット単位の符号語を連続化した符号語列を、再生データ列に復調する復調装置であって、前記符号語列を 6 ビット毎の符号語に再構成する手段と、後続の符号語が前記複数の符号化テーブルのうち、どの符号化テーブルで符号化がなされるかを示す判定情報と、後続の符号語とを基にして、前記符号語列を再生データ列に復調する手段とを有することを特徴とする復調装置。

【請求項 13】

請求項 6 乃至請求項 10 の何れか 1 項に記載の変調装置を用いて符号化がなされた符号語が少なくとも一部記録されている事を特徴とする情報記録媒体。

【請求項 14】

請求項 1 乃至請求項 5 の何れか 1 項に記載の変調方法を用いて符号化がなされた符号語を伝送情報として情報伝送を行う事を特徴とする情報伝送方法。

【請求項 15】

請求項 6 乃至請求項 10 の何れか 1 項に記載の変調装置を用いて符号化がなされた符号語を伝送情報として情報伝送を行う事を特徴とする情報伝送装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、変調方法、変調装置、復調方法、復調装置、情報記録媒体、情報伝送方法および情報伝送装置に関し、特にデジタル情報信号を、 $(1, k)$ ラン・レンジス・リミテッド（以下、「 $(1, k)$  RLL」と記す）制限で、 $k=7$ 以上で 12 以下の何れかなる制限をもつ記録符号系列によって光ディスクや磁気ディスクなどの記憶媒体に記録するためにデジタル情報信号を、 $(1, k)$ ラン・レンジス・リミテッド（以下、「 $(1, k)$  RLL」と記す）制限で、 $k=7$ 以上で 12 以下の何れかなる制限をもつ情報符号系列を変調、復調、記録、伝

送するのに好適な変調方法、変調装置、復調方法、復調装置、情報記録媒体、情報伝送方法および情報伝送装置に関するものである。

【0002】

【従来の技術】

従来より、光ディスクあるいは磁気ディスクなどの記録媒体に、一連のデジタル情報信号を記録するための記録変調方式としては、(1, 7) RLLがよく使われている。しかし従来から使われている(1, 7) RLLでは、直流(DC)付近の信号成分抑圧が困難であり、ビットパターンによっては大きなDC成分を生じ、例えば、サーボ信号帯域に情報信号成分のスペクトルが混入し、サーボ性能に悪影響が及ぶ問題が生ずる事が予想される。

【0003】

これに対して、特開平6-195887号公報「記録符号変調装置」では、特定ビットパタンの繰り返しを防止する事で、DC成分の抑圧を図るための提案がなされている。また、特開平10-340543号公報「エンコード装置、デコード装置、エンコード方法、及びデコード方法」では、(1, 7) RLL規則を乱さないように冗長ビットを挿入することで、DC成分の抑圧を図るための提案がなされている。

【0004】

あるいは、特開2000-105981公報「データ変換方式および装置」によれば、(1, 8) RLL規則による8/12変調を用いて、最大ラン長を(1, 7) RLL規則に比べ符号語数に余裕を持たせ、この余裕分をDC成分の抑圧制御に用いる提案がなされている。

【0005】

【発明が解決しようとする課題】

然るに、特開平6-195887号公報によると、ビット反転や、ランダムイズ等の手段によって特定パタンの繰り返しの低減は図れるものの、十分にDC成分の抑圧をすることは困難である。また、特開平10-340543号公報によれば、DC成分の抑圧は前者に比べれば大きいものの、冗長ビットの挿入による記録容量の低下が生じてしまう。特開2000-105981公報によれば、D

C成分の抑圧が冗長ビット無しに図れるものの、12ビットの符号化テーブルを複数必要とし、符号化規則が複雑になるという問題点を有していた。

## 【0006】

本発明は上記の問題点に鑑みてなされたもので、冗長ビットを用いること無しに(1, k) RLL規則で、 $k = 7$ 以上で12以下の何れかの制限下におけるDC成分の抑圧を4ビットを6ビットに変換可能な符号化テーブルを用いて達成しようとするものである。

## 【0007】

## 【課題を解決するための手段】

本発明は上述した問題点を解決するために 4ビット単位の入力データ語を6ビット単位の出力符号語に符号化するための変換をする際に、前記各入力データ語に対応する前記各出力符号語と、次の前記入力データ語を符号化するために使用される符号化テーブルを指定する符号化テーブル指定情報とを含む複数の符号化テーブル参照し、前記各出力符号語は2進数の出力符号語列として順次直接結合して(1, k) RLL (ラン・レンジス・リミテッド) 規則で $k = 7$ 以上で12以下の何れかを満足する出力符号語として出力することを特徴とする変調方法を提供する。

## 【0008】

また、本発明は上述した問題点を解決するために4ビット単位の入力データ語を6ビット単位の出力符号語に符号化する変換手段を有し、前記変換手段は、前記入力データ語を前記出力符号語にそれぞれ符号化するための符号化テーブルを複数備えており、前記各符号化テーブルのそれぞれには前記各入力データ語に対応する前記各出力符号語と、次の前記入力データ語を符号化するために使用される符号化テーブルを指定する符号化テーブル指定情報とを含み、前記各出力符号語は2進数の出力符号語列として順次直接結合して(1, k) RLL (ラン・レンジス・リミテッド) 規則で $k = 7$ 以上で12以下の何れかを満足する出力符号語として出力することを特徴とする変調装置を提供する。

## 【0009】

また、本発明は上述した問題点を解決するために上述した変調方法を用いて符



号化された6ビット単位の符号語を連続化した符号語列を、再生データ列に復調する復調方法であって、前記符号語列を6ビット毎の符号語に再構成し、後続の符号語が前記複数の符号化テーブルのうち、どの符号化テーブルで符号化がなされるかを示す判定情報と、後続の符号語とを基にして、前記符号語列を再生データ列に復調することを特徴とする復調方法を提供する。

## 【0010】

また、本発明は上述した問題点を解決するために上述した変調装置を用いて符号化された6ビット単位の符号語を連続化した符号語列を、再生データ列に復調する復調装置であって、前記符号語列を6ビット毎の符号語に再構成する手段と、後続の符号語が前記複数の符号化テーブルのうち、どの符号化テーブルで符号化がなされるかを示す判定情報と、後続の符号語とを基にして、前記符号語列を再生データ列に復調する手段とを有することを特徴とする復調装置を提供する。

## 【0011】

また、本発明は上述した問題点を解決するために上述した変調装置を用いて符号化がなされた符号語が少なくとも一部記録されている事を特徴とする情報記録媒体を提供する。

## 【0012】

また、本発明は上述した問題点を解決するために上述した変調方法を用いて符号化がなされた符号語を伝送情報として情報伝送を行う事を特徴とする情報伝送方法を提供する。

## 【0013】

また、本発明は上述した問題点を解決するために上述した変調装置を用いて符号化がなされた符号語を伝送情報として情報伝送を行う事を特徴とする情報伝送装置を提供する。

## 【0014】

## 【発明の実施の形態】

以下、図1～図9を参照して、本発明の変調に関する実施形態を説明する。図1は本発明の変調装置の基本構成図、図2は本発明の変調装置のブロック構成図、図3は図2に示す符号化部周辺のブロック構成図、図4は図2に示す変調装置

の符号化動作を説明するためのフローチャート、図 5 は本発明の変調装置による RLL (1, 7) 規則を満たすための DSV 制御を説明するためのフローチャート、図 6 本発明の変調装置による RLL (1, 8) 規則を満たすための DSV 制御を説明するためのフローチャート。図 8 は本発明の変調装置に用いられる 4 つの符号化テーブル " S (k) = 0 " ~ " S (k) = 3 " の各内容を表す図であり、S (k) はテーブルの状態、D (k) は入力データ語、C (k) は出力符号語であり、デシマルとバイナリの表記をしてある。また S (k + 1) は次にとるテーブルを示す状態である。

## 【 0 0 1 5 】

さて、(1, 7) RLL あるいは (1, 8) RLL 制限を満足する 6 ビット単位の出力符号語の種類は図 7 のようになる。この符号語種類を基にした符号化テーブルの一例としては、図 8 に示すような 4 つの符号化テーブル (符号化テーブル番号 S (k) = " 0 " ~ " 3 ") が構成できる。S (k) = " 0 " ~ S (k) = " 3 " は、4 つの符号化テーブルにそれぞれ割り当てられた符号化テーブル選択番号を表す。また、図 8 中の S (k + 1) は、次の符号化を行うために用いる符号化テーブルを選択する符号化テーブル選択番号を表す。なお、データ語 D (k) と符号語 C (k) との割り当ては符号化規則を乱さずかつ復調に支障をきたさないよう配置を変えることは可能である。例えば、図 1 5 に示す符号化テーブルは図 8 のテーブルのデータ語 D (k) と符号語 C (k) の割り当てを換えた配置をしており、このようにデータ語 D (k) と符号語 C (k) との割り当てを符号化規則を乱さないよう配置換えが可能であり、本発明の実施は図 8 の符号化テーブルの構成以外でも本発明は有効である。

## 【 0 0 1 6 】

また、本発明による DSV の制御則を満たしながら、例えば 8 ビットのデータ語を 12 ビットの符号語ビットに割り当てるように、4 の整数倍のビットからなるデータ語を 6 の整数倍の符号ビットに変換する符号化テーブルの構成は、本発明から容易に類推ができ、本発明に含まれる事は明らかである。

## 【 0 0 1 7 】

まず図 1 を用いて、本発明の変調装置 1 について説明をする。変調がなされる

べき画像、音声等を図示せぬ離散化手段によってバイナリ系列に変換されたデジタル情報信号はフォーマット部 1 1 で誤り訂正符号の付加やセクタ構造化等のいわゆるフォーマット化がなされたのち 4 ビットごとのソースコード系列となり 4 - 6 変調器 1 2 に加えられる。

## 【 0 0 1 8 】

4 - 6 変調器 1 2 は一例として図 8 に示した符号化テーブル 1 3 を用いて後述の符号化処理を行うとともに所定の同期語を付加したのち、NRZ I 変換回路 1 4 にて NRZ I 変換して記録信号として記録駆動回路 1 5 に送出し、記録媒体 2 に記録あるいは伝送符号化手段 3 1 にて伝送符号化がなされ伝送媒体 3 に送出される。

## 【 0 0 1 9 】

図 2 は図 1 の 4 - 6 変調部 1 2 について、より詳細に動作を説明するための構成例を示したブロック図である。入力データ語（ソースコード） $D(k)$  は符号語選択肢有無検出回路 1 2 1 と符号化テーブルアドレス演算部 1 2 2、同期語生成部 1 2 3 にそれぞれ加えられる。符号語選択肢有無検出回路 1 2 1 では  $D(k)$  と状態  $S(k)$  を用いて  $DSV$  極性の異なる符号語候補があるかどうかを検出する。この検出結果と  $D(k)$  とを基に符号化テーブルアドレス演算がなされ複数の符号化テーブル 1 3 から符号化候補を  $C(k)0$ 、 $C(k)1$  として前者を符号語メモリ「0」1 2 4、後者を符号語メモリ「1」1 2 5 に送出する。

## 【 0 0 2 0 】

符号語メモリ「0」1 2 4、符号語メモリ「1」1 2 5 には  $DSV$  演算メモリ「0」1 2 6、 $DSV$  演算メモリ「1」1 2 7 が接続され、符号語  $C(k)0$ 、 $C(k)1$  が符号語メモリ「0」1 2 4、符号語メモリ「1」1 2 5 に入力される毎に  $CDS$  の計算を行い記憶されている  $DSV$  値の更新する。ここで、符号語選択肢有無検出回路 1 2 1 によって選択肢があるソースコード  $D(k)$  が検出された場合、絶対値比較部 1 2 8 によって、 $DSV$  メモリ「0」1 2 6、 $DSV$  メモリ「1」1 2 7 に蓄えられている  $DSV$  の絶対値が比較され、メモリ制御部 1 2 9 で  $DSV$  の絶対値が小さい符号語メモリに蓄えられた符号語を選択し出力符号語として外部出力するとともに選択されなかった符号語メモリ、 $DSV$  演算メ

モリの内容を、選択した符号語メモリ、DSV演算メモリの内容に入れ替える。

#### 【0021】

図3が図2の符号化テーブル周辺を詳細に示した図であり、図4が以上述べた内容を詳細に示すフローチャートである。なお、本説明では符号語メモリを2つにし、符号語選択肢有無検出回路121で選択肢をもつ $D(k)$ が検出された場合、すぐに出力符号語を出す場合について説明をしたが、符号語メモリは2つに限られたものではなく、選択肢をもつ $D(k)$ が検出された場合、すぐに出力符号語を出す必要はなく、さらに何個かのメモリをもち、選択可能なソースコードをいくつか見て一番DSVの小さな符号語列を選択出力する方法でも本発明は有効である。図3において、最大ラン長設定130は、(1, 7)RLLに制限をするかあるいは $k$ を8以上に制限をするかの制御信号を、符号語選択肢有無検出回路に出力する手段であり、動作の詳細は後述する。また、同図において、最小ラン繰り返し検出131は最短反転の繰り返し数を監視する手段であり、動作の詳細は後述する。

#### 【0022】

つぎに図9を用いて4ビット単位の入力データ語 $D(k)$ を(1, 7)RLL制限による符号化する場合について具体的に説明する。入力データ語 $D(k)$ 、 $D(k+1) \cdots$ として「4, 5, 6, 7, 8 (デシマル)」を例として用いる。符号化の初期状態では、説明を省略する同期語の挿入などの操作によって、符号化テーブルの初期選択番号を決定し、例えば、符号化テーブル $S(k) = "0"$ が選択される。この符号化テーブル $S(k) = "0"$ に、入力データ語 $D(k) = 4$ を入力すると、出力符号語 $C(k) = 18$  (デシマル)が出力され、また、次の符号化テーブル選択番号 $S(k+1) = "1"$ が選択される。次に、選択された符号化テーブル $S(k) = "1"$ に、入力データ語 $D(k) = 5$ を入力すると、出力符号語 $C(k) = 2$  (デシマル)が出力され、また、次の符号化テーブル選択番号 $S(k+1) = "2"$ が選択されることになる。以下同様に、符号化テーブル $S(k) = "2"$ に入力データ語 $D(k) = 6$ を入力すると、出力符号語 $C(k) = 18$ が出力され、符号化テーブル選択番号 $S(k+1) = "3"$ が選択され、次に符号化テーブル $S(k) = "3"$ に入力データ語 $D(k) =$

7を入力すると、出力符号語 $C(k) = 21$ が出力され、符号化テーブル選択番号 $S(k+1) = "0"$ が選択され、そして、符号化テーブル $S(k) = "0"$ に入力データ語 $D(k) = 8$ を入力すると、出力符号語 $C(k) = 21$ が出力され、符号化テーブル選択番号 $S(k+1) = "1"$ が選択されることになる。

## 【0023】

この結果、入力データ語 $D(k)$ として「4, 5, 6, 7, 8 (デシマル)」は出力符号語 $C(k)$ として「010010, 000010, 010010, 010101, 010101 (バイナリ)」に符号化されて順次出力される。従って、前記した5つの出力符号語 $C(k)$ を順次直接結合した一連の出力符号語列は、

010010000010010010010101010101

となり、(1, 7) RLLの制限を満足する出力符号語列を得ることができる。

## 【0024】

この例では選択肢が存在するソースコードが出現をしていないがこのように、図1から図3に示した変調装置によって、図8になる符号化テーブルを用いることで4ビットごとのソースコード $D(k)$ とひとつ前の符号語を出力した際に出力された $S(k+1)$ を1ワード(ソースコードでの4ビット長)遅延させた $S(k)$ とによって、(1, 7) RLL制限を満足する符号語列を順次直接結合する事によって得ることができる。

## 【0025】

つぎに図5を用いて符号語選択肢有無検出回路121の動作について詳細に説明をする。図5が(1, 7) RLLの場合の選択肢有無演算回路121がなす動作についてフローチャートに示したものである。ステップ201の条件1についてみると、ひとつ前に符号化がなされた符号語 $C(k-1)$ のLSB側のゼロランを検出し4の場合(ステップ201でYesの場合)、すなわち図8の符号化テーブルで $C(k-1)$ がバイナリで010000のとき、 $S(k)=3$ で、 $D(k)$ が0~3の場合(条件1-1、ステップ202でYesの場合)には $C(k)0$ として $S(k)=3$ のテーブルから符号語を選択し、 $C(k)1$ として $S(k)=1$ の符号語を選択して“選択肢有り”という検出信号を選択肢有無検出回路121から出力する(ステップ206)。 $S(k)$

$k)=2$ で $D(k)$ が7以上の時（条件1-2、ステップ203でYesの場合）、 $C(k)0$ として $S(k)=2$ のテーブルから符号語を選択し、 $C(k)1$ として $S(k)=1$ の符号語を選択して“選択肢有り”という検出信号を選択肢有無検出回路121から出力する（ステップ207）。ステップ201、ステップ202およびステップ203でそれぞれNoの場合は、 $C(k)0$ 、 $C(k)1$ とも $D(k)$ 、 $S(k)$ で選択された符号語“選択肢なし”（ステップ208）として判断を終了する。

## 【0026】

同様に、条件2（ステップ204）では、 $C(k-1)$ のLSB側のゼロランが5のとき、あるいは条件3（ステップ205）では $C(k-1)$ のLSB側のゼロランが1か2の時には図5のフローチャートに従った判断によって選択肢があるかどうかを検出する。

## 【0027】

ステップ204の条件2についてみると、ひとつ前に符号化がなされた符号語 $C(k-1)$ のLSB側のゼロランを検出し5の場合（ステップ204でYesの場合）、すなわち図8の符号化テーブルで $C(k-1)$ がバイナリで100000のとき、 $S(k)=3$ で、 $D(k)$ が0~1の場合（条件2-1、ステップ209でYesの場合）には $C(k)0$ として $S(k)=3$ のテーブルから符号語を選択し、 $C(k)1$ として $S(k)=1$ の符号語を選択して“選択肢有り”という検出信号を選択肢有無検出回路121から出力する（ステップ210）。 $S(k)=2$ で $D(k)$ が10以上の時（条件2-2、ステップ211でYesの場合）、 $C(k)0$ として $S(k)=2$ のテーブルから符号語を選択し、 $C(k)1$ として $S(k)=1$ の符号語を選択して“選択肢有り”という検出信号を選択肢有無検出回路121から出力する（ステップ212）。ステップ204、ステップ209およびステップ211でそれぞれNoの場合は、 $C(k)0$ 、 $C(k)1$ とも $D(k)$ 、 $S(k)$ で選択された符号語“選択肢なし”（ステップ208）として判断を終了する。

## 【0028】

ステップ205の条件3についてみると、ひとつ前に符号化がなされた符号語 $C(k-1)$ のLSB側のゼロランを検出し1か2の場合（ステップ205でYesの場合）、すなわち図8の符号化テーブルで $C(k-1)$ がバイナリで010010、01

0100、000010、000100、001010、100100、101010あるいは100010のとき、 $S(k)=2$ で、 $D(k)$ が0~1の場合（ステップ213でYesの場合）には $C(k)0$ として $S(k)=2$ のテーブルから符号語を選択し、 $C(k)1$ として $S(k)=0$ の符号語を選択して“選択肢有り”という検出信号を選択肢有無検出回路121から出力する（ステップ214）。ステップ205およびステップ213でそれぞれNoの場合は、 $C(k)0$ 、 $C(k)1$ とも $D(k)$ 、 $S(k)$ で選択された符号語“選択肢なし”（ステップ208）として判断を終了する。

## 【0029】

ステップ215の条件4についてみると、ひとつ前に符号化がなされた符号語 $C(k-1)$ のLSB側のゼロランを検出し、1の場合（ステップ215でYesの場合）、010010、000010、001010、101010あるいは100010のとき、 $S(k)=2$ で $D(k)$ が12か13すなわちバイナリで101010のとき（ステップ216でYesの場合）、次に接続される符号語のMSB（最上位ビット）が1の場合（ステップ217でYesのとき）には $C(k)0$ として $S(k)=2$ の符号語、 $C(k)1$ として $S(k)=0$ の符号語を選択して“選択肢有り”という検出信号を選択肢有無検出回路121から出力する（ステップ218）。ステップ215、ステップ216およびステップ217でそれぞれNoの場合は、 $C(k)0$ 、 $C(k)1$ とも $D(k)$ 、 $S(k)$ で選択された符号語“選択肢なし”（ステップ208）として判断を終了する。

## 【0030】

さて、 $C(k-1)$ が010000で $S(k)=3$ で $D(k)$ が3以下の場合、 $S(k)=1$ の符号語と交換が可能で有る事はどちらを選んでも最大の0の連続が7に収まり、（1，7）RL規則を乱すことがないことが明らかであり、また、 $C(k-1)$ が010000の場合は次を取る符号語が $S(k)=2$ または3で符号化がなされることが符号化テーブル13によって限定されており、かつ $S(k)$ が1，2，3の符号化テーブル13に含まれる符号語は各々独立しているすなわち同じ符号語が存在していないことから復号時に問題が生じることはない。

## 【0031】

同様に、 $C(k-1)$ が100000すなわちLSB側のゼロランが5のときも同様に

(1, 7) RLL規則を乱すことがなく、さらに復号時の問題は生じない。

【0032】

C(k-1)のLSB側のゼロランが1か2の符号語は次に $S(k) = 1$ または2または3を取る符号語であり、 $S(k) = 0$ の符号化テーブルに含まれる符号語は $S(k) = 2$ または3に含まれる符号語と同じ符号語が存在している。しかし、 $S(k) = 0$ の符号語のうち、 $D(k) = 0$ または1の符号語である000001は他のテーブルに存在しないユニークな符号語であり、 $S(k) = 2$ の符号語と交換をしても復号時の問題は生じない。

【0033】

同様に、C(k-1)のLSB側のゼロランが1の符号語は次に $S(k) = 1$ または2または3を取る符号語であり、 $S(k) = 0$ の符号化テーブルに含まれる符号語は $S(k) = 2$ または3に含まれる符号語と同じ符号語が存在している。しかし、 $S(k) = 0$ の符号語のうち、 $D(k) = 12$ または13の符号語である000000は他のテーブルに存在しないユニークな符号語であり、かつぎの符号語の最上位ビットが1であれば、 $k = 7$ が維持でき、 $S(k) = 2$ の符号語と交換をしても復号時の問題は生じない。

【0034】

以上説明をしたように、図5に従った符号語の交換によってDSVの制御ができることは交換される符号語に含まれる1の偶奇が異なることから説明ができる。すなわち、C(k-1)が010000で、 $S(k) = 3$ で $D(k) = 0$ だった場合、C(k)0は101001であり、C(k)1は001001である。NRZI変換する際の直前の極性が1だったとすると、前者は001111であり、最終ビットが1なので0となる一方、後者は111000であり最終ビットが1なので1になる。図10にこの様子を示す。a)が前者でありb)が後者である。上段がC(k-1)、C(k)、C(k+1)であり、下段がNRZI変換後の符号語である。図10から明らかなように、C(k)を交換することでNRZI変換後の極性が変わりDSV値が変化をする。よってDSVの小さくなるようなパターンを選択することによってDC成分の抑圧ができるのである。

【0035】



次に図6を用いて(1, 8) RLL制限を持つ符号語の変調法について説明を行う。(1, 7) RLLか(1, 8) RLLかは図3の最大ラン長設定130によって決められるかあるいは初期設定からどちらかに決めておく。また、(1, 8) RLLの場合の符号化テーブルは図8の(1, 7) RLLと同様の符号化テーブルが使用できる。

## 【0036】

さて、(1, 8) RLLの場合は最大ラン長が(1, 7) RLLより1ビット長くゆるされているので条件が図5と比較をして異なってくる。図6中、条件1では $C(k-1)$ のLSB側のゼロランが4か5の時(ステップ301でYesの場合)、 $S(k)=3$ のテーブルが選択され、かつ $D(k)$ が0~3の場合(条件1-1、ステップ302でYesの場合)、 $C(k)0$ に $S(k)=3$ の符号語、 $C(k)1$ に $S(k)=1$ の符号語が選択可能である(ステップ303)。また、LSB側のゼロランが4か5の時(ステップ301でYesの場合)、 $S(k)=2$ のテーブルが選択され、かつ $D(k)$ が7以上の場合(条件1-2、ステップ304でYesの場合)、 $C(k)0$ に $S(k)=2$ の符号語、 $C(k)1$ に $S(k)=1$ の符号語が選択可能である(ステップ305)。ステップ301、ステップ302およびステップ304でそれぞれNoの場合は、 $C(k)0$ 、 $C(k)1$ とも $D(k)$ 、 $S(k)$ で選択された符号語“選択肢なし”(ステップ306)として判断を終了する。

## 【0037】

同様に、条件2では $C(k-1)$ のLSB側のゼロランが1の時(ステップ307でYesの場合)、 $S(k)=2$ が選択された場合、 $D(k)=12$ か13であれば(ステップ308でYesの場合)、 $C(k)0$ には $S(k)=2$ の符号語、 $C(k)1$ には $S(k)=0$ の符号語が選択可能である(ステップ309)。ステップ307およびステップ308でそれぞれNoの場合は、 $C(k)0$ 、 $C(k)1$ とも $D(k)$ 、 $S(k)$ で選択された符号語“選択肢なし”(ステップ306)として判断を終了する。

## 【0038】

また、条件3では $C(k-1)$ のLSB側のゼロランが3以下のとき(ステップ310でYesの場合)、 $S(k)=2$ で $D(k)$ が0または1の時(ステップ311でYesの場合)、 $C(k)0$ には $S(k)=2$ の符号語、 $C(k)1$ には $S(k)=0$ の符号語が選択可能

である（ステップ312）。テップ310およびステップ311でそれぞれNoの場合は、 $C(k)0$ 、 $C(k)1$ とも $D(k)$ 、 $S(k)$ で選択された符号語“選択肢なし”（ステップ306）として判断を終了する。

#### 【0039】

条件4では $C(k-1)$ のLSB側のゼロランが2のとき（ステップ313でYesの場合）、 $S(k)=2$ で $D(k)$ が12か13の場合（ステップ314でYesの場合）、次に選択される符号語のMSBが1の場合（ステップ315でYesの場合）、 $C(k)0$ には $S(k)=2$ の符号語、 $C(k)1$ には $S(k)=0$ の符号語が選択可能である（ステップ316）。ステップ313、314および315で何れかNoの場合には $C(k)0$ 、 $C(k)1$ とも $D(k)$ 、 $S(k)$ で選択された符号語“選択肢なし”（ステップ306）として判断を終了する。

#### 【0040】

以上説明をしたように図6の条件判断に従えば、(1, 8)RL規則を満たしたDC成分の抑圧がなされた符号語の生成が可能である。

#### 【0041】

$k=9$ のRL規則の符号化の場合には、ステップ301の判断にゼロランが6の場合が加わり、 $k=9$ が満足される場合、ステップ303、ステップ305が実行される。また、ステップ315の判断が不要になり、つぎの符号語がいずれの場合でも $C(k)0$ には $S(k)=2$ の符号語、 $C(k)1$ には $S(k)=0$ の符号語が選択可能になる。

#### 【0042】

$k=10$ のRL規則の符号化の場合には、図6のステップ301の判断にゼロランが6の場合が加わり、さらにステップ313で、ゼロランが3も選択が可能になり、ステップ315の判断が不要になり、つぎの符号語がいずれの場合でも $C(k)0$ には $S(k)=2$ の符号語、 $C(k)1$ には $S(k)=0$ の符号語が選択可能になる。

#### 【0043】

$k=11$ の場合にはさらにステップ313で、ゼロランが4の場合も選択が可

能になり、ステップ 3 1 5 の判断が不要になり、つぎの符号語がいずれの場合でも  $C(k)0$  には  $S(k) = 2$  の符号語、 $C(k)1$  には  $S(k) = 0$  の符号語が選択可能になる。

## 【 0 0 4 4 】

$k = 12$  の場合にはさらにステップ 3 1 3 で、ゼロランが 5 の場合も選択が可能になり、ステップ 3 1 5 の判断が不要になり、つぎの符号語がいずれの場合でも  $C(k)0$  には  $S(k) = 2$  の符号語、 $C(k)1$  には  $S(k) = 0$  の符号語が選択可能になる。

## 【 0 0 4 5 】

さて、以上説明したように、本発明になる符号化テーブルを用いることによって  $(1, 7)$  RLL 制限あるいは  $k = 8$  以上で 12 以下のいずれかの RLL 制限を持つ符号生成可能な変調方法、あるいは変調装置を実現が可能である。

## 【 0 0 4 6 】

なお、以上述べた DSV 制御を踏まえて、4 ビットのデータ語を 6 ビットの符号語に変換する変調方法あるいは変調装置はあらかじめ選択可能なビットパターン複数備え、例えば 8 ビットのデータ語を 12 ビットの符号語あるいは 4 の整数倍のビットからなるデータ語を 6 の整数倍ビットからなる符号語に変換する符号化テーブルを構成することは容易であり、本発明に含まれる。

## 【 0 0 4 7 】

つぎに図 2 乃至図 4 を用いて DSV 制御の方法について上述した符号語の選択をふまえた説明を加える。説明では図 5 に示した  $(1, 7)$  RLL の変調過程を用いるが、 $k = 8$  以上で 12 以下のいずれかの RLL 制限の符号語でも図 6 に示したように選択肢があるかどうかの判断をすることによって同様に DSV 制御が可能である。

## 【 0 0 4 8 】

まず、図 4 において、初期テーブル設定 (ステップ 1 0 1) は符号語に付加される同期語等の後続の  $S(k)$  を決定することで設定が可能である。次に 4 ビットのソースコード  $D(k)$  を入力し (ステップ 1 0 2)、 $S(k)$  と  $D(k)$  とによって図 8 の符号化テーブルに従って符号化を行う。この過程でひとつ前に符

号化した $C(k-1)$ を見てLSB側のゼロラン長を演算し、符号語の選択肢があるかどうかを図5の条件に従って判断をする(ステップ103)。なお、図2、図3では $C(k-1)$ が符号出力の手段から入力されているが、ひとつ前の入力データと、状態 $S(k)$ を保持することによって求めることも可能である。

## 【0049】

符号化テーブルに選択可能符号語が存在しない場合(ステップ103で「しない」場合)は符号語メモリ「0」124、符号語メモリ「1」125に符号化テーブルから出力された符号語を $C(k)0$ 、 $C(k)1$ として(ステップ107)それぞれ符号語メモリ「0」124、符号語メモリ「1」125に付加してCDSを演算し、DSVメモリ126、DSVメモリ127を更新する(ステップ108)。

## 【0050】

符号化テーブルに選択可能符号語が存在する場合(ステップ103で「する場合)、選択肢が存在することを示す信号を符号語選択肢有無検出回路121から出力し、DSVメモリ0、1の絶対値を絶対値演算回路によって演算をし、符号語メモリから絶対値の小さい符号系列を出力手段から出力する(ステップ104)。その後、選択した符号語系列に選択しなかった符号語メモリの内容を入れ替えると同時にDSV演算メモリを採用した値に採用しなかった値を入れ替える(ステップ105)。その後、図5、図6の説明で述べたように、符号語候補として選択が可能な符号語を $S(k)$ で決定される一方の符号化テーブルと他方の符号化テーブルから選択をし $C(k)0$ 、 $C(k)1$ として出力する(ステップ106)。その後、符号語メモリ「0」124、符号語メモリ「1」125に符号化テーブルから出力された符号語を $C(k)0$ 、 $C(k)1$ として(ステップ107)＜符号語候補 $C(k)0$ 、 $C(k)1$ それぞれについてCDSを計算し、DSVメモリ「0」、「1」を更新し、符号語メモリ「0」、「1」に $C(k)0$ 、 $C(k)1$ を付加し、DSVメモリ126、DSVメモリ127を更新する(ステップ108)。

## 【0051】

以上の操作を符号化の終了(ステップ109)まで行うことによってDC成分

が抑圧された符号語の生成が終了する。

【0052】

次に本発明になる最短ビットの反転が連続した場合のビット操作について説明をする。最短ビットの反転は伝送路の周波数特性が低い時に位相同期をかりにくくする場合があり、このような伝送路について、本発明では次に述べるような手段によって最短ビット反転の連続を阻止することが可能である。

【0053】

図8の符号化テーブルによれば、最短ビット反転の連続は010101の繰り返しまたは101010の繰り返しとによって発生する。010101の繰り返しは $S(k) = 0$ または $S(k) = 3$ ののち、 $D(k) = 7$ が連続した場合に生じる。このときは、最小ラン繰り返しカウンタによって $S(k) = 0$ で、 $D(k) = 7$ ののち、例えば $D(k+1) = 7$ 、 $D(k+2) = 7$ の場合、 $D(k+1) = 13$ を選択する。本来この符号語は $S(k+2) = 3$ の符号語であるので、 $C(k+1) = 000000$ の後、RL規則を乱さないように $S(k) = 1$ の符号語を選ぶことで010101の繰り返しが発生したことを検出し、復号することが可能である。

【0054】

例えば $k = 9$ から12の場合には、 $S(k) = 0$ で、 $D(k) = 7$ ののち、例えば $D(k+1) = 7$ 、 $D(k+2) = 7$ となる場合、 $D(k+1) = 13$ を選択して( $C(k) = 000000$ を選択して) $S(k+2) = 1$ を出力する。このとき、 $D(k+2)$ は7なので $C(k+2) = 4$ すなわちバイナリで00100が出力される。復号時は000000ののち、000100を検出した場合に最小ランの繰り返し制限による符号語の入れ替えが生じた事を認識し、 $D(k+1)$ 、 $D(k+2)$ とも正常に復号することが可能である。 $k = 8$ の場合には、例えば、 $D(k+2)$ を10から15の何れかにかえ同様に復号することが可能である。

【0055】

さて、101010の繰り返しの場合は $S(k) = 2$ で $D(k) = 12$ のとき、符号語は101010で、 $S(k+1) = 2$ となり、その後、 $D(k+1) =$

1 2 の時、符号語は 1 0 1 0 1 0 で  $S(k+2) = 2$  で  $D(k+2) = 1 2$  で 1 0 1 0 1 0 の符号語が出力される。この場合は  $S(k+1)$  を 0 に変えることによって 1 0 1 0 1 0 が 0 0 0 0 0 0 に交換することができ、後に述べる復調方法によって問題なく復調することが可能である。以上述べたように、本発明によれば、最小反転の繰り返しを阻止することが可能である。

## 【 0 0 5 6 】

図 3 を用いて以上の動作を再度説明する。最小ラン繰り返し監視 1 3 1 は  $S(k)$  と  $D(k)$  を監視しながら最小反転の繰り返しが起こる  $D(k)$  と  $S(k)$  の繰り返しをカウント（最小ラン繰り返しカウント）をする。この情報を符号語選択肢有無検出回路に送出し、最小ランの繰り返しを上述の手段によって阻止する。

## 【 0 0 5 7 】

さらに、符号語選択肢有無検出回路には最大ラン長設定 1 3 0 が接続されており、最大ランの設定すなわち (1, 7) R L L の変調を行うか、あるいは  $k = 8$  以上 1 2 以下うち、何れかの変調を行うかの設定をする。この設定は図示せぬシステムコントローラ等の手段によって切り替えることが可能である。

## 【 0 0 5 8 】

つぎに本発明による復調方法と復調装置について説明をする。図 1 1 は本発明に好適な復調装置の実施の一例である。入力符号語のビット列は NRZI 復調手段 5 0 1 で NRZI 復調され、同期検出回路 5 0 2 によって同期語が検出され、NRZI 復調された信号および同期語はパラレル 6 ビットに変換するためのタイミング信号であるワードクロックによってシリアル／パラレル変換器 5 0 3 によって 6 ビットごとの符号列  $C(k)$  に構成される。こののちワードレジスタ 5 0 4 に入力され 1 ワード遅延がなされた符号語  $C(k-1)$  は符号語の判定情報の検出装置 5 0 5 に入力され後述の判定情報が演算出力される。判定情報と入力符号語  $C_k$  は状態演算器 5 0 6 に入力され 4 つの符号化テーブルのうちどの符号化テーブルによって符号化がなされたかを示す状態  $S(k)$  を出力し、アドレス生成部 5 0 7 にて  $C(k-1)$  と  $S(k)$  とに指定されるアドレスにより例えば図 1 2 に示す復号テーブル 5 0 8 から出力データ語が出力される。

## 【 0 0 5 9 】

判定情報は図 1 2 に示すように 0, 1, 2 の 3 つの場合わけがなされ LSB 側のゼロラン長によって次の符号語がどの符号化テーブルによって符号化がなされるのかを示すものである。すなわち、ひとつ前の符号語  $C(k-1)$  と現在の符号語がどの符号化テーブルで符号化がなされているのかを知ることによって  $C(k-1)$  が  $D(k-1)$  に復調される。

## 【 0 0 6 0 】

(式 1)

```

if (判定情報 == 0) [
    if (C(k)が0の符号化テーブルにある符号語)
        S(k)=0;
    elseif (C(k)が1の符号化テーブルにある符号語)
        S(k)=1;]
if (判定情報 == 1) [
    if (C(k)が1の符号化テーブルにある符号語)
        S(k)=1;
    elseif (C(k)が2の符号化テーブルにある符号語)
        S(k)=2;
    elseif (C(k)が3の符号化テーブルにある符号語 || 1)
        S(k)=3;
    elseif (C(k) == 0 && C(k-1) == 32)
        S(k)=3;
    elseif (C(k) == 0 && C(k-1) == 42)
        S(k)=2;]
if (判定情報 == 2) [
    if (C(k)が3の符号化テーブルにある符号語 || 9 || 5 || 2)
        S(k)=3;
    elseif (C(k)が2の符号化テーブルにある符号語 || 1 0 || 8)
        S(k)=2;

```

```

elseif (C(k)==21)
    S(k)=0;]
if (D(k-1)==2) [
    if (C(k)が4)
        D(k-1)は7;
    ]

```

式1がC(k)と判定情報とからS(k)を求めるための演算であり、C言語によって記述されている。本演算によれば、判定情報とC(k)、C(k-1)とからS(k)が求まり、図13の復調テーブルによってCk-1をDk-1に復調可能である。なお、本演算では(1, 7) RLLの場合、k=8から12の場合さらに、k=9より大の最小ラン長の制限を設けた場合についてすべての復調演算を含んでいる。このため、(1, 7) RLLでもk=8から12の場合についてもDSVの制御方法すなわち図5、図6のいずれを選んだ場合でも復調装置は同一のもので正常に復調がなされる。

#### 【0061】

たとえば、図14のように010000 001001 000001 000101 010001 なる符号語列が図11に示す復調装置に入力された時、C(k-1)=010000の判定情報はLSB側のゼロラン長が4であることから図12のように、判定情報は2である。また、次の符号語C(k)が001001(デシマルで9)と続いており、式1の最初の条件判定に当てはまるからS(k)は3であることがわかる。よって図13の復調テーブルのC(k-1)で、010000の行のS(k)が3であることから、D(k-1)として15と求まる。すなわち、k時点のC(k)が生成された符号化テーブルの状態情報(番号)S(k)からk-1時点のC(k-1)に対応するD(k-1)が復号されるのである。同様にして001001は判定情報が0であり、続く符号語の000001は符号化テーブルのS(k)=0にあるため、図13の復調テーブルによってD(k-1)は0と求まる。同様にして000001はD(k-1)が1、000101はD(k-1)が2と求まる。なお、001001はDSV制御のために図5の条件1-1で交換がなされた符号語であるが、正常に復号が



できていることが以上の説明によって明らかである。

【 0 0 6 2 】

【発明の効果】

以上説明したように本発明によれば、連続する2進数のデータ系列を4ビット単位の入力データ語に変換した後に、(1, 7) RLL規則または $k = 8$ 以上12以下の何れかのRLL規則を満足する6ビット単位の出力符号語列に変換が可能であり、また、出力符号語列に冗長ビットを加えることなくDSV制御が可能であるから、出力符号語列のDC成分の効果的な抑圧が可能である変調装置とその復調装置を提供することができるという利点を有する。

【図面の簡単な説明】

【図 1】

本発明の変調装置の基本構成図である。

【図 2】

本発明の変調装置のブロック構成図である。

【図 3】

図 2 に示す符号化部周辺のブロック構成図である。

【図 4】

図 2 に示す変調装置の符号化動作を説明するためのフローチャートである。

【図 5】

本発明の(1, 7) RLLの場合のDSV制御を示すフローチャートである。

【図 6】

本発明の(1, 8) RLLの場合のDSV制御を示すフローチャートである。

【図 7】

4ビット単位のデシマル入力データ語に対応する6ビット単位のバイナリ出力符号語を表す図である。

【図 8】

本発明の変調装置に用いられる4つの符号化テーブル $S(k) = 0 \sim S(k) = 3$ の各内容を表す図である。

【図 9】

本発明の変調装置における符号化過程を説明する図である。

【図 1 0】

本発明の変調装置の動作を説明するための図である。

【図 1 1】

本発明の復調装置の実施例のブロック図である。

【図 1 2】

本発明の復調装置に用いられる判定情報を示す図である。

【図 1 3】

本発明の復調装置に用いられる復調テーブルを示す図である。

【図 1 4】

本発明の復調装置の動作を説明するための図である。

【図 1 5】

本発明の符号化テーブルの他の例を示す図である。

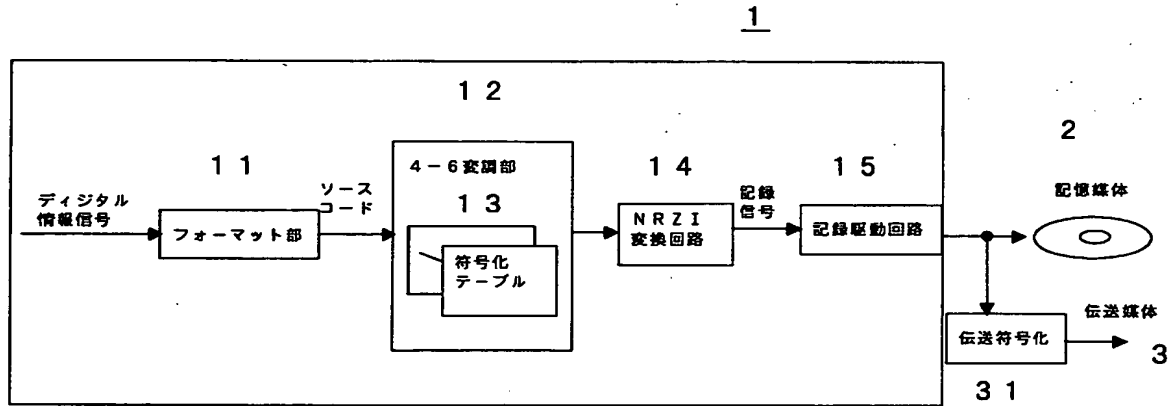
【符号の説明】

- 1 …変調装置、
- 2 …記録媒体、
- 3 …伝送媒体、
- 1 1 …フォーマット部、
- 1 2 … 4 - 6 変調部、
- 1 3 …符号化テーブル、
- 1 4 …NRZI 変換回路、
- 1 5 …記録駆動回路、
- 3 1 …伝送符号部、
- 1 2 1 …符号語選択肢有無検出回路、
- 1 2 2 …符号化テーブルアドレス演算部
- 1 2 3 …同期語生成部、
- 1 2 6、1 2 7 …DSV 演算メモリ、
- 1 2 4、1 2 5 …符号語メモリ、
- 1 2 8 …絶対値比較部、

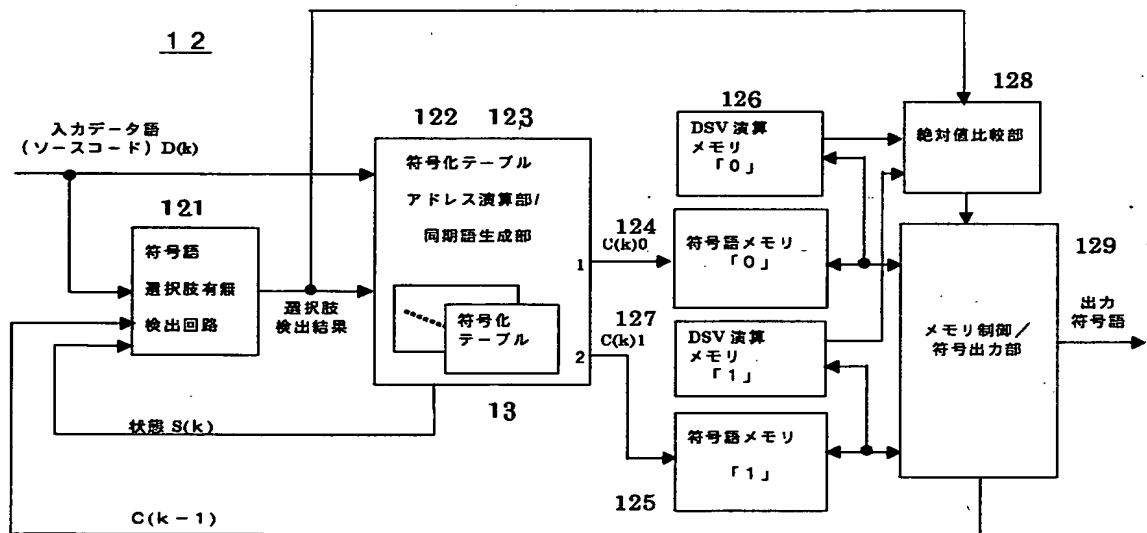
- 1 2 9 …メモリ制御符号化出力部、
- 5 0 1 …NRZ I 復調、
- 5 0 2 …同期検出回路、
- 5 0 3 …シリアル／パラレル変換器、
- 5 0 4 …ワードレジスタ、
- 5 0 5 …符号語判定情報検出装置、
- 5 0 6 …状態演算器、
- 5 0 7 …アドレス生成部、
- 5 0 8 …復号テーブル、

【書類名】 図面

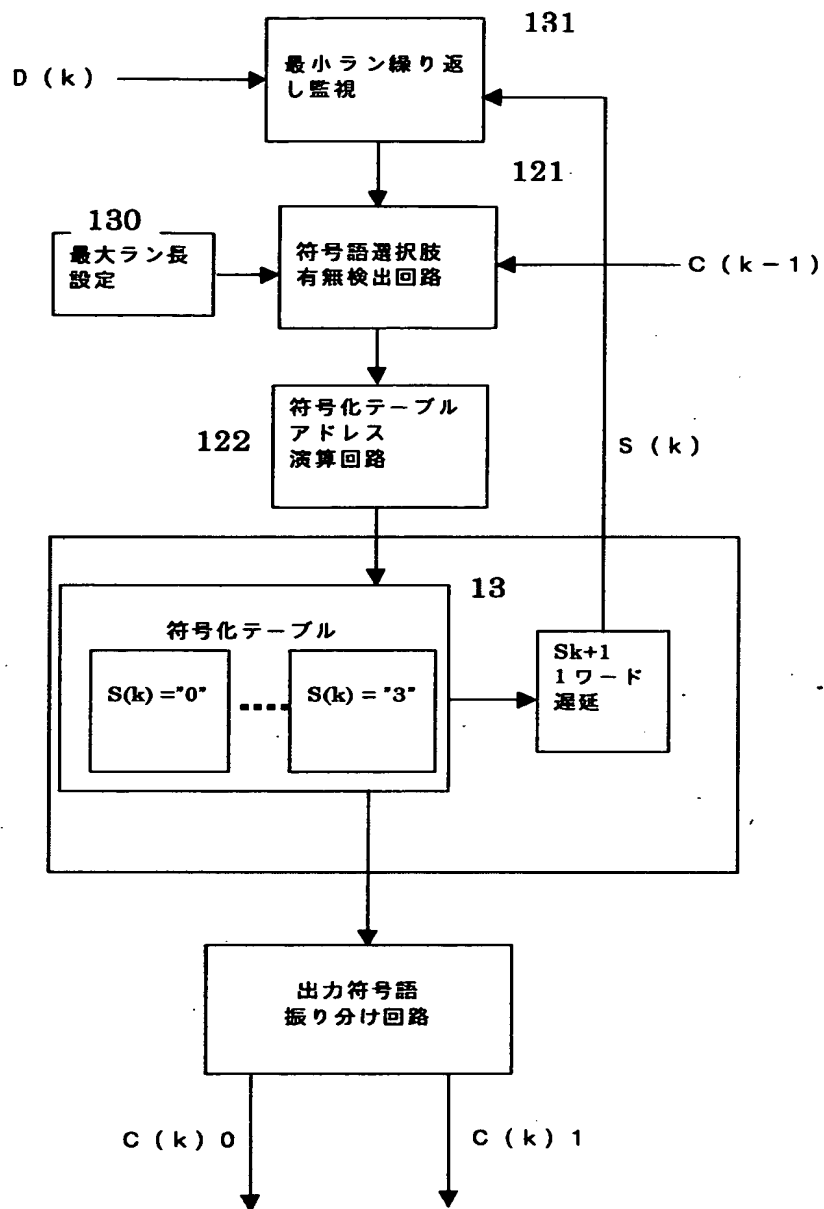
【図 1】



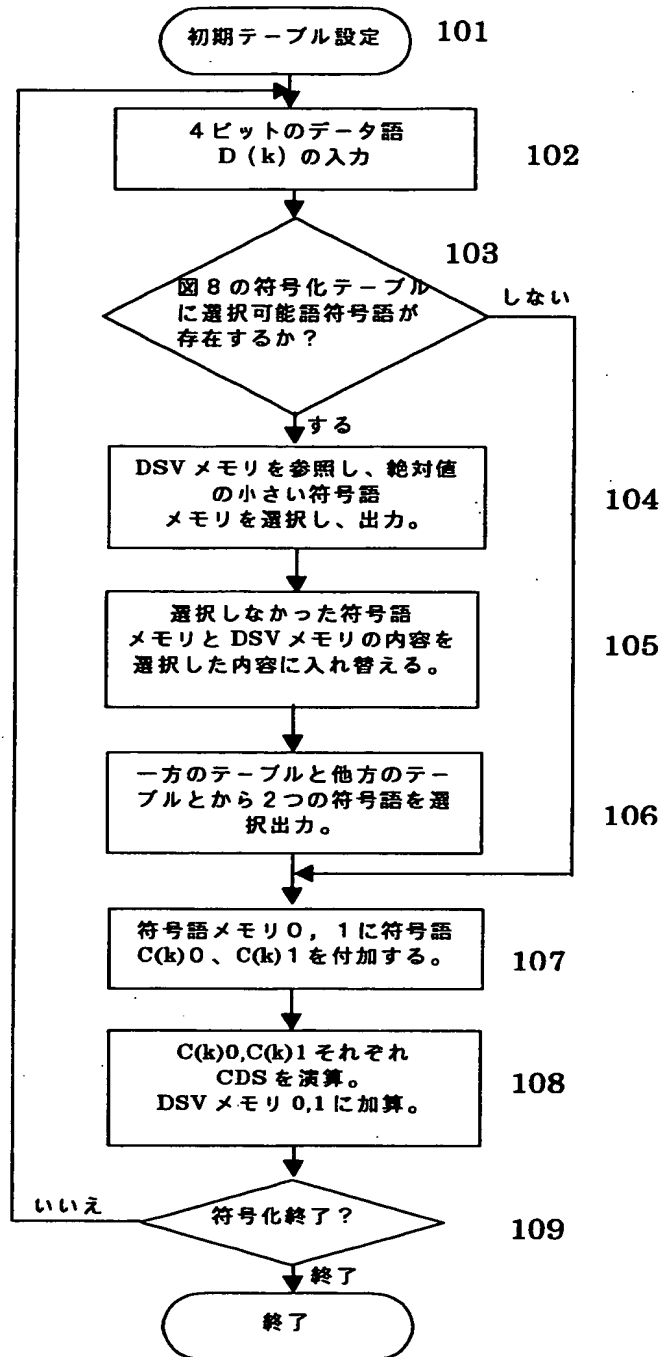
【図 2】



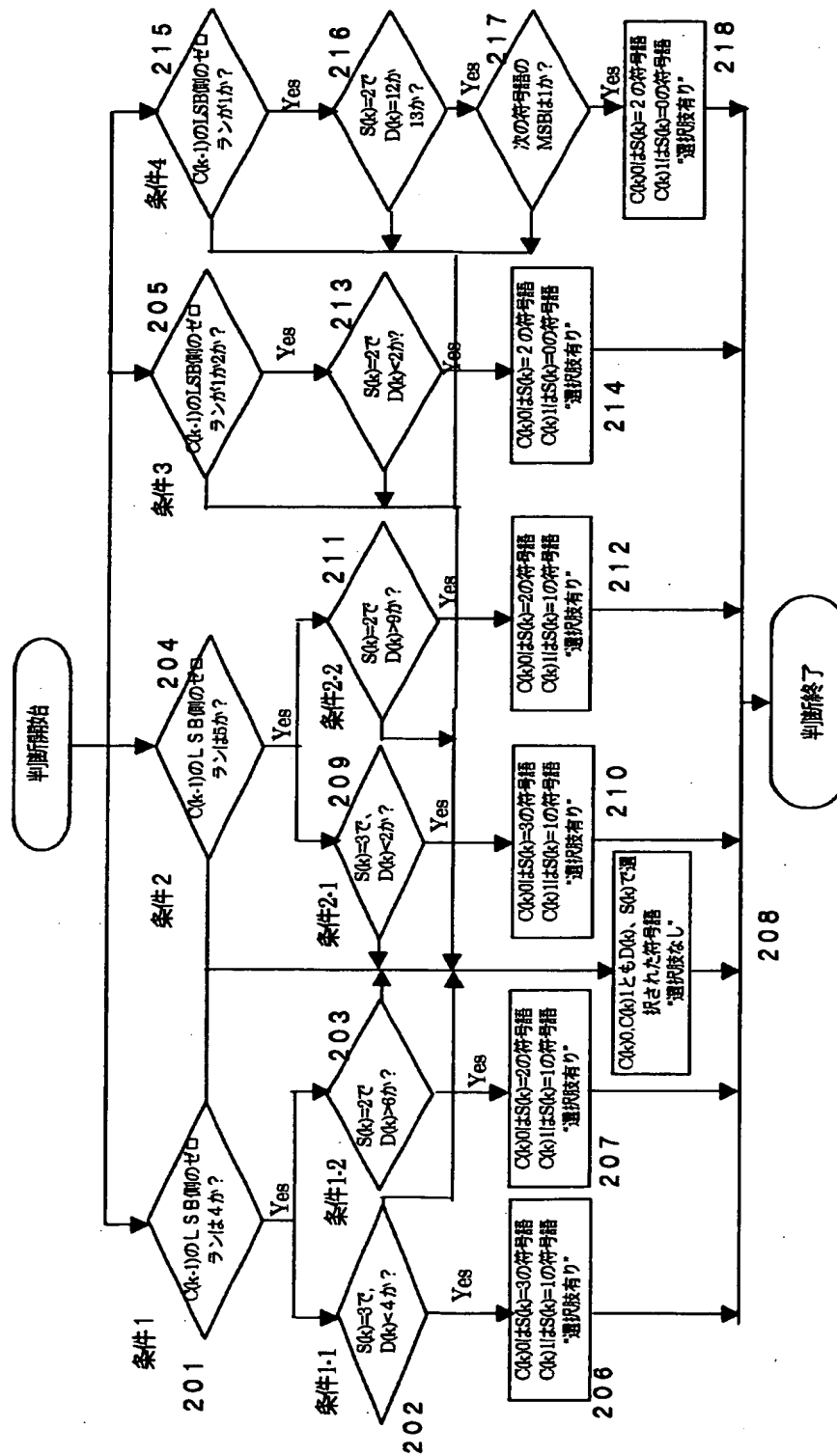
【図3】



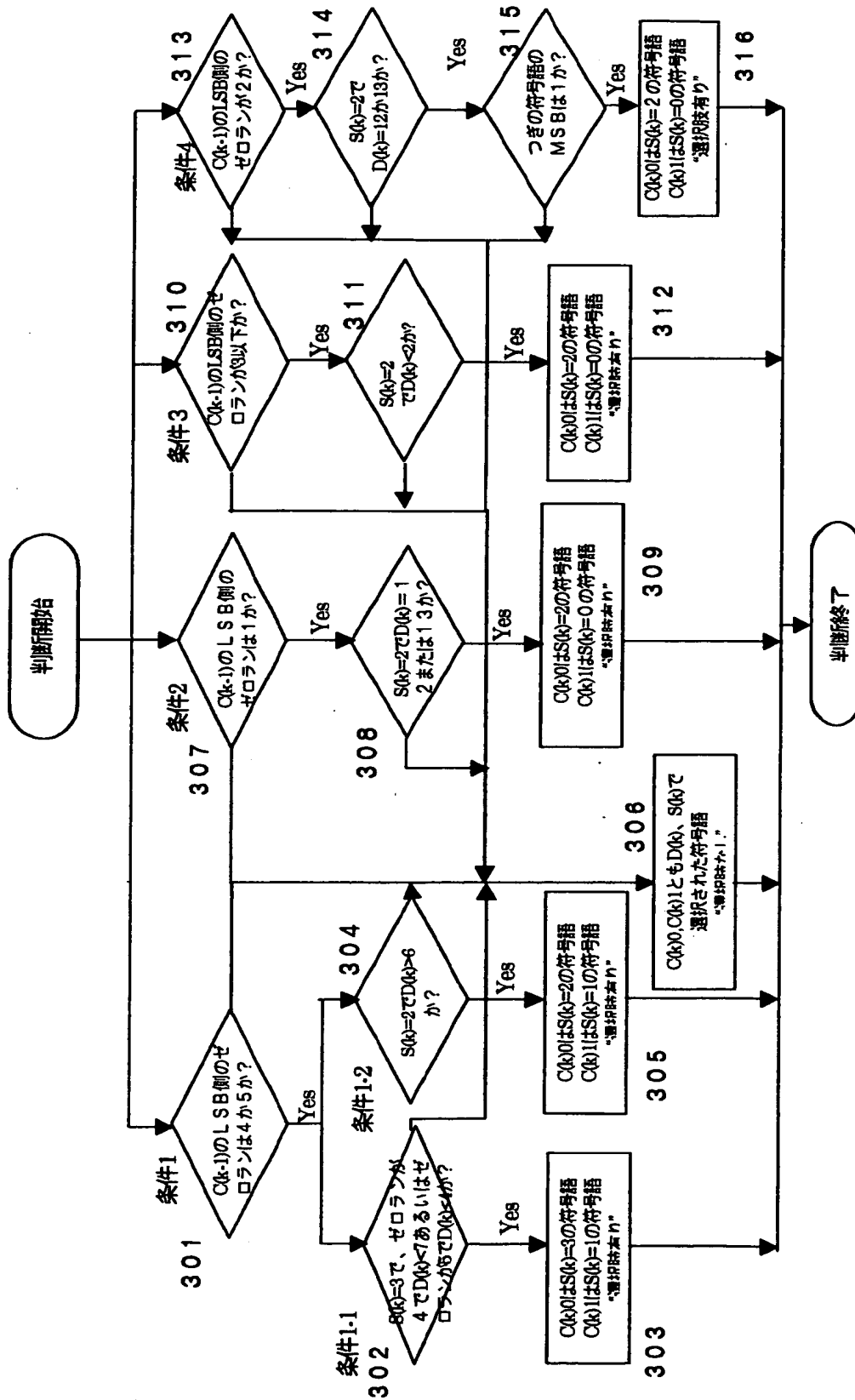
【図4】



【图 5】



【图 6】





【図 7】

デシマル	バイナリ
0	000000
1	000001
2	000010
4	000100
5	000101
8	001000
9	001001
10	001010
16	010000
17	010001
18	010010
20	010100
21	010101
32	100000
33	100001
34	100010
36	100100
37	100101
40	101000
41	101001
42	101010

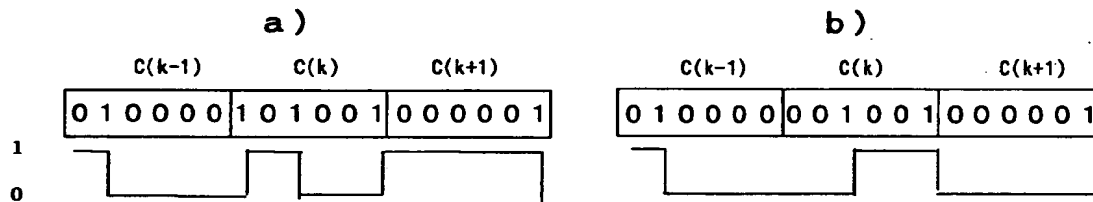
【図 8】

S(k) D(k)	0			1			2			3		
	C(k)	S(k+1)		C(k)	S(k+1)		C(k)	S(k+1)		C(k)	S(k+1)	
0	1 000001	0	9	001001	0	33	100001	0	41	101001	0	
1	1 000001	1	9	001001	1	33	100001	1	41	101001	1	
2	17 010001	0	5	000101	0	17	010001	0	37	100101	0	
3	17 010001	1	5	000101	1	17	010001	1	37	100101	1	
4	18 010010	1	2	000010	1	18	010010	1	34	100010	1	
5	18 010010	2	2	000010	2	18	010010	2	34	100010	2	
6	18 010010	3	2	000010	3	18	010010	3	34	100010	3	
7	21 010101	0	4	000100	1	36	100100	1	21	010101	0	
8	21 010101	1	4	000100	2	36	100100	2	21	010101	1	
9	20 010100	1	4	000100	3	36	100100	3	20	010100	1	
10	20 010100	2	10	001010	1	42	101010	1	20	010100	2	
11	20 010100	3	8	001000	1	40	101000	1	20	010100	3	
12	0 000000	2	10	001010	2	42	101010	2	32	100000	2	
13	0 000000	3	10	001010	3	42	101010	3	32	100000	3	
14	16 010000	2	8	001000	2	40	101000	2	16	010000	2	
15	16 010000	3	8	001000	3	40	101000	3	16	010000	3	

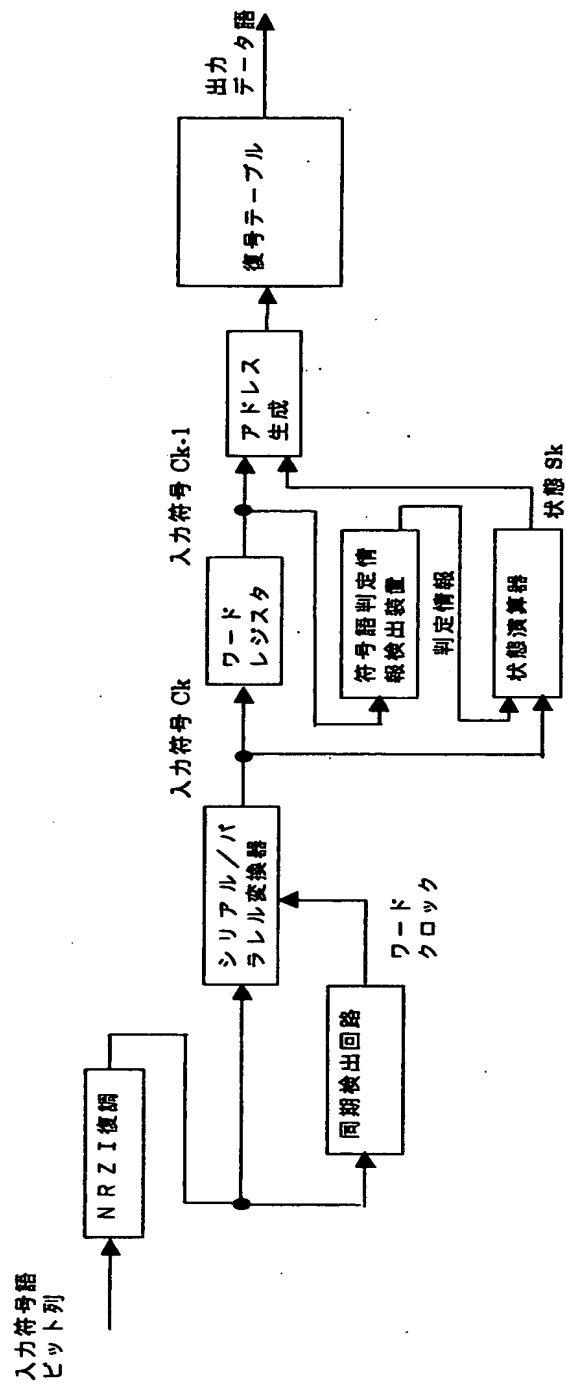
【図9】

入力データ語	テーブル選択情報	出力符号語	テーブル選択情報
D(k)	S(n)	C(k)	S(n+1)
4	0	18	1
5	1	2	2
6	2	18	3
7	3	21	0
8	0	21	1

【図10】



【図 11】



【図 1 2】

判定情報	LSB側のゼロラン長	次の状態
0	0	0
1	1, 2, 3,	1, 2, 3,
2	4, 5, 6	2, 3

【図 1 3】

Ck-1		判定情報	Dk-1 / Sk			
デシマル	バイナリ		0	1	2	3
0	000000	2	7	-	12	13
1	000001	0	0	1	-	-
2	000010	1	-	4	5	6
4	000100	1	-	7	8	9
5	000101	0	2	3	-	-
8	001000	1	-	11	14	15
9	001001	0	0	1	-	-
10	001010	1	-	10	12	13
16	010000	2	-	-	14	15
17	010001	0	2	3	-	-
18	010010	1	-	4	5	6
20	010100	1	-	9	10	11
21	010101	0	7	8	-	-
32	100000	2	-	-	12	13
33	100001	0	0	1	-	-
34	100010	1	-	4	5	6
37	100101	0	2	3	-	-
40	101000	1	-	11	14	15
41	101001	0	7	8	-	-
42	101010	1	-	10	12	13

【図 1 4】

Dk	Ck	判定情報	Sk
15	010000	2	3
0	001001	0	0
1	000001	0	1
2	000101	0	0
3	010001	0	-

【図 15】

S(k)	0				1				2				3			
	D(k)	C(k)	S(k+1)		D(k)	C(k)	S(k+1)		D(k)	C(k)	S(k+1)		D(k)	C(k)	S(k+1)	
0	1	000001	0	9	001001	0	33	100001	0	41	101001	0				
1	17	010001	1	5	000101	1	17	010001	1	37	100101	1				
2	18	010010	2	2	000010	2	18	010010	2	34	100010	2				
3	17	010001	0	5	000101	0	17	010001	0	37	100101	0				
4	18	010010	1	2	000010	1	18	010010	1	34	100010	1				
5	1	000001	1	9	001001	1	33	100001	1	41	101001	1				
6	18	010010	3	2	000010	3	18	010010	3	34	100010	3				
7	20	010100	1	4	000100	1	36	100100	1	20	010100	1				
8	21	010101	0	4	000100	2	36	100100	2	21	010101	0				
9	20	010100	2	4	000100	3	36	100100	3	20	010100	2				
10	21	010101	1	10	001010	1	42	101010	1	21	010101	1				
11	20	010100	3	8	001000	1	40	101000	1	20	010100	3				
12	16	010000	2	8	001000	2	40	101000	2	16	010000	2				
13	0	000000	3	10	001010	3	42	101010	3	32	100000	3				
14	16	010000	3	8	001000	3	40	101000	3	16	010000	3				
15	0	000000	2	10	001010	2	42	101010	2	32	100000	2				